CLAIMS

1. A method comprising:

receiving a plurality of events;

applying the plurality of events to a correlation function, wherein the correlation function is implemented as a state machine; and

generating a specific event if the correlation function is satisfied by the plurality of events.

2. A method as recited in claim 1 wherein the correlation function is a class object.

3. A method as recited in claim 1 further including:

receiving a data element; and

applying the data element and at least one of the plurality of events to the correlation function.

4. A method as recited in claim 1 further including:

receiving a plurality of data elements; and

applying the plurality of data elements and the plurality of events to the correlation function.

5. A method as recited in claim 1 further including communicating the specific event to at least one event consumer that subscribed to the specific event.

6. A method as recited in claim 1 further including continuing to receive additional events and apply the additional events to the correlation function if the correlation function is not satisfied by the plurality of events.

7. A method as recited in claim 1 further including resetting the correlation function after generating a specific event.

8. A method as recited in claim 1 further including:

creating an instance of a particular state machine; and

defining transitions for the particular state machine by subscribing to at least one event.

9. A method as recited in claim 8 further including applying an update consumer to the particular state machine to update the state of the particular state machine.

10. One or more computer-readable memories containing a computer program that is executable by a processor to perform the method recited in claim 1.

11. A method comprising:

receiving a plurality of events;

receiving a plurality of data elements;

identifying a plurality of correlation functions;

applying the plurality of events and the plurality of data elements to the plurality of correlation functions; and

generating a specific event if at least one of the plurality of correlation functions is satisfied.

12. A method as recited in claim 11 wherein each of the plurality of correlation functions is implemented as a state machine.

13. A method as recited in claim 11 wherein each of the plurality of correlation functions is an instance of a class object.

14. A method as recited in claim 11 further including communicating the specific event to at least one event consumer that subscribed to receive the specific event.

15. A method as recited in claim 11 further including:

receiving additional events;

receiving additional data elements; and

applying the plurality of events, the additional events, the plurality of data elements and the additional data elements to the plurality of correlation functions.

16. A method as recited in claim 11 further including:

receiving additional events;

receiving additional data elements;

receiving additional correlation functions; and

applying the plurality of events, the additional events, the plurality of data elements and the additional data elements to the plurality of correlation functions and the additional correlation functions.

17. A method as recited in claim 16 further including generating the specific event if at least one of the plurality of correlation functions or at least one of the additional correlation functions is satisfied.

18. A method as recited in claim 11 wherein the specific event generated is dependent on which correlation function is satisfied.

19. One or more computer-readable memories containing a computer program that is executable by a processor to perform the method recited in claim 11.

20. A method comprising:

identifying a schema for creating state machines, the state machines to correlate at least two events;

creating an instance of a particular state machine;

defining transitions for the particular state machine by subscribing to at least one event; and

applying an update consumer to the particular state machine to update the state of the particular state machine.

21. A method as recited in claim 20 further including deleting the particular state machine if the particular state machine reaches a final state.

22. A method as recited in claim 20 wherein the particular state machine includes a timer, the method further including deleting the particular state machine if the timer expires.

23. A method as recited in claim 20 wherein the particular state machine correlates at least one event and at least one data element.

24. A method as recited in claim 20 wherein the particular state machine correlates a plurality of events and a plurality of data elements.

25. A method as recited in claim 20 further including determining a current state of the particular state machine.

26. A method as recited in claim 20 wherein the particular state machine is an instance of a class object.

27. One or more computer-readable memories containing a computer program that is executable by a processor to perform the method recited in claim 20.

28. An apparatus comprising:

a plurality of event consumers; and

an event correlator coupled to the plurality of event consumers, the event correlator to receive events from at least one event source and to receive data elements from at least one data source, the event correlator further to receive at least one correlation function and to apply the received events and the received data elements to the correlation function, wherein the event correlator generates a specific event if the received events and the received data satisfy the correlation function.

29. An apparatus as recited in claim 28 wherein the event correlator communicates the specific event to the plurality of event consumers.

30. An apparatus as recited in claim 28 wherein the event correlator communicates the specific event to event consumers that have requested to receive the specific event.

31. An apparatus as recited in claim 28 wherein the event correlator communicates the specific event to a plurality of filters, wherein each of the plurality of filters is associated with one of the plurality of event consumers.

32. An apparatus as recited in claim 28 wherein the event correlator includes at least one state machine to implement the correlation function.

33. An apparatus as recited in claim 28 wherein the event correlator includes at least one state machine to implement the correlation function, wherein the event correlator identifies a current state of each state machine.

34. An apparatus as recited in claim 28 wherein the event correlator continues to receive additional events and additional data elements and apply the additional events and the additional data elements to the correlation function.

35. One or more computer-readable media having stored thereon a computer program that, when executed by one or more processors, causes the one or more processors to:

receive a plurality of events;

identify a plurality of correlation functions;

apply the plurality of events to the plurality of correlation functions to determine whether any of the plurality of correlation functions are satisfied by the plurality of events; and

generate a specific event if one of the plurality of correlation functions is satisfied by the plurality of events.

36. One or more computer-readable media as recited in claim 35 wherein the plurality of correlation functions are implemented as state machines.

37. One or more computer-readable media as recited in claim 35 wherein each of the plurality of correlation functions is implemented as a state machine, and wherein the state machine is a class object.

38. One or more computer-readable media as recited in claim 37 further causing the one or more processors to identify a current state of the state machine.

39. One or more computer-readable media as recited in claim 35 further causing the one or more processors to:

create a new instance of a state machine to implement a particular correlation function; and

define transitions for the new instance of the state machine by subscribing to at least one event.